UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/563,568	10/30/2006	Volker Sauermann	2058.091US1	3664

21186 7590 11/26/2010

SCHWEGMAN, LUNDBERG & WOESSNER, P.A.
P.O. BOX 2938
MINNEAPOLIS, MN 55402

EXAMINER	
VU, TUAN A	
ART UNIT	**PAPER NUMBER**
2193	
NOTIFICATION DATE	**DELIVERY MODE**
11/26/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

uspto@slwip.com
request@slwip.com

Office Action Summary	Application No.	Applicant(s)	
	10/563,568	SAUERMANN ET AL.	
	Examiner	Art Unit	
	TUAN A. VU	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 06 January 2006.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-8 and 10-23 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-8, 10-23 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on 06 January 2006 is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date 1/06/06;8/03/10.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to the application filed 1/06/2006.

Claims 1-8, 10-23 have been submitted for examination.

Claim Objections

2. Claims 10 and 15 are objected to because of the following informalities: the word 'forth' is clearly a typographic impropriety that will be treated more correctly as 'fourth'. Appropriate correction is required.

Claim Rejections - 35 USC § 101

3. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

> The Federal Circuit has recently applied the practical application test in determining whether the claimed subject matter is statutory under 35 U.S.C. § 101. The practical application test requires that a " useful, concrete, and tangible result" be accomplished. An "abstract idea" when practically applied is eligible for a patent. As a consequence, an invention, which is eligible for patenting under 35 U.S.C. § 101, is in the "useful arts" when it is a machine, manufacture, process or composition of matter, which produces a concrete, tangible, and useful result. The test for practical application is thus to determine whether the claimed invention produces a "useful, concrete and tangible result".

The current focus of the Patent Office in regard to statutory inventions under 35 U.S.C. § 101 for method claims and claims that recite a judicial exception (software) is that the claimed invention recite a practical application. Practical application can be provided by a physical transformation or a useful, concrete and tangible result. The following link on the World Wide Web is for the United States Patent And Trademark Office (USPTO) policy on 35 U.S.C. §101.

<http://www.uspto.gov/web/offices/pac/dapp/opla/preognotice/guidelines101_20051026.pdf>

4. Claims 10-15, 17-23 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

5. Specifically, claim 10 recites a product comprising a counter, first, second, third, and forth retriever; which according to the Disclosure are elements that cannot be construed as hardware components included with the claim as a whole to realize the acts of determining or retrieving data of the claim. The claim amounts to listing of software functionalities

("Functional Descriptive Material' deficiency – see Annex IV, Guidelines pg. 44-45) and cannot constitute a permissible category of subject matter. Hence, claim 10 along with claims 11-15 are rejected for non-statutory of subject matter.

6. Claim 17 recites a system including variables and a evaluator for comparing; hence amounts to software functionalities without hardware to realize actions of the software. Claim 17 along with claims 18-23 are rejected for non-statutory subject matter for the same reasons set forth above.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 1-8, 10-23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ming-Chuan Wu, "Encoded Bitmap Indexes and Their Use for Data Warehouse Optimization", D17, Darmstadt Dissertation, January 2001, (herein Wu_D17).

As per claim 1, Wu_D17 discloses a computer implemented method for automatic software tuning, the method comprising:

calculating at least one threshold value (A ≤ 864 – sec 4.2.1 – pg. 95; A op v – Algorithm 4 - pg. 97; significance threshold – pg. 115 – Note: operator <, >, ≤, ≥ used in predicate reads threshold being calculated to enforce predicate A for a *selection* task and indexing on A; see: *i* ≤ *w* - limited buffer Algorithm 7 pg. 111) for at least one parameter (e.g. *bitmap index, attributes, cust_id, nation_id, orders_id* – Fig. 4.3; grouping attributes, dimension tables – pg. 89-90; *result*

set, group-set bitmap index – pg. 92; *set of orders_id* - pg. 93; group set Fig. 4.12, Fig. 4.14 pg. 108-110; response time, θ – pg. 114; sec 4.21; Algorithm 4 pg. 97; Algorithm 7, pg. 111; $0< \theta < 0.5$, significance threshold – pg. 115; hits – pg. 125-126) influencing the performance of a software application (*query processing, performance* – sec 4.1, pg. 85; Fig. 4.1, 4.2, Fig. 86-87) with regards to a specific task (Selections, Group bys, Aggregations Order bys – pg.86 top; sec 4.3.1 pg. 105-120; Aggregation, Algorithm 8 pg. 112; limited buffer $i \leq w$, Algorithm 7 pg. 111);

comparing the at least one threshold value to at least one corresponding current value (Algorithm 4, Algorithm 11 - from above; $0< \theta < 0.5$, pg.115 – Note: operator for predicate and for buffer limit or upper limit for response time reads on comparing actual variable value with its upper limit).

Wu_D17 does not explicitly disclose selecting an algorithm from a plurality of algorithms for performing the task in accordance with the result of the comparing step.

Wu_D17 discloses significance threshold to evaluate significance and response between two scenarios in order to determine which scenario is to be selected with the algorithm for Aggregation (Choose Alternative 1, Choose Alternative 2 – pg. 116) hence has either disclosed selecting an algorithm based on evaluation significance of a threshold, and comparing against upper limit of the threshold or would have rendered the selection obvious. Based on Wu_D17 reconsideration of the what alternative to use for a algorithm in view of the threshold and response time relationship as above, the selection of algorithm based on readjusting of the threshold is more recognized in Wu_D17 correlating results with requirement of bound-partitioning scenario/alternative to determine it a rough partitioning or poor decision being made as a result of such bound (see C.2 pg. 201-202), where the threshold can be modified. It would

have been obvious for one of ordinary skill in the art to implement the algorithm and selection of partitioning alternative in Wu_D17 so that after comparing value against a threshold, and evaluating results from partitioning using the required bound/threshold, a determination would be made to change to another algorithm or partitioning alternative in order to improve response time of the algorithm based on modifying the range or required bounds of the threshold; because a smaller or less ambitious range bounded by a rough threshold, as shown above, would yield better response time.

As per claim 2, Wu_D17 discloses (by virtue of the rationale in claim 1 and based on see sec 4.3.2 pg. 115; C.2 pg. 201-202):

measuring the performance of the selected algorithm; checking whether the measured performance complies with the at least one threshold value; and recalculating the at least one threshold value in case of non-compliance; all of which for the same reasons as set forth above.

As per claim 3, Wu_D17 discloses wherein the at least one threshold value separates the value range of the parameter into at least two intervals of a first dimension (Algorithm 9: *s, t, r,* θ – pg. 115-116 Note: threshold included in inequality expression to represent a partitioning requirement reads on two intervals of first dimension - e.g. estimate cost *r, s, t*).

As per claims 4-5, Wu_D17 discloses (in view of claim 1 rationale)

wherein the selecting selects the algorithm that is assigned to the interval that includes the corresponding current value of the first dimension;

wherein at least one further threshold value separates the value range of a further parameter into at least two intervals of a second dimension (see C.2 pg. 201-202; sums of the parts, picky-backed- pg. 118-119; sec 4.1.2 rewriting the number of groups – Note: reorganize

large group into smaller subgroup as picky-backed requirements reads on selecting a better response time scenario based on separating upper bound and lower bound of ranges or dimension table or group-set for a complex Aggregation task).

As per claim 6, Wu_D17 discloses modifying ranges into subset to improve response time (see pg. 117-119) via picky-backing of subrange representing by smaller bitmap coverages, and dividing a threshold having a fixed ends into 3 subranges (see $0<s/t< \theta$; $\theta \leq s/t \leq 1-\theta$; $1-\theta <s/t <1$ - C.2 pg. 201-202); hence the overlapping effect of subdividing range with a $\leq$ operator teaches first dimension (or range/interval) of a current parameter would intersect dimension of the second parameter of a second dimension(or range/interval). According to this overlapping of range related to a task (selections, group bys), Wu_D17 further discloses improvement in partition selection (i.e. indexing and bounding) with employing pipelining of overlapping operations to overcome the overhead cost when analyzing performance of tasks (selections, group bys) for improving response time (e.g. reduces the total processing time … overlapping – sec 2.1.2 pg. 21); hence the concept of overlapping interval of index in group/partition selection

Wu_D17 does not explicitly disclose wherein the selecting selects the algorithm that is assigned to the intersection of the interval of the first dimension that includes the corresponding current parameter value of the first dimension and the interval of the second dimension that includes the corresponding current parameter value of the second dimension.

Based on Wu_D17's selecting of Aggregation alternative by which sub-grouping are used to improve response time and sub-dividing of threshold for finding a better partitioning response as set forth above using a overlapping operator, and the overhead cost in performance due to box bounding as set forth above in the overlapped pipelining approach, it would have

been obvious for one of ordinary skill in the art to implement Wu_D17's selection of best alternative so that the subdivision of threshold requirement or regrouping of dimension set to that sub-divided set of range intersect with each other in terms that intersection intervals including portion of the first dimension containing the corresponding current parameter value of the first dimension and portion of the second dimension containing the corresponding current parameter value of the second dimension, because overlapping regions or range – as indicated in Wu_D17's overlapping approach - would cover for all the partitions included in a larger range whose upper or lower bound are to be excluded as shown in Wu_D17's readjusting of threshold (see C.2 pg. 201-202) such that all data or parameter values (response time, return of group hits – see pg. 117-119) to be considered within the large range (group-set, dimension table/set) would be all inclusive in the final account and ensuing evaluation of result leading to choosing the best algorithm (e.g. bitmap indexed dynamic Aggregation scenario – see sec 2.1.2 pg. 21) as to improve overhead cost in support of best selection of partition/group as endeavored in Wu_D17's pipelining of operations via overlapping of index intervals from above.

As per claim 7, Wu_D17 disclose wherein each threshold value (bounds of a range- sec 3.5 pg. 77-79: range selection, RangeEval Opt Appendix C.1) corresponds to a break-even point (sec 3.5.3 pg. 79-80) where two neighbouring algorithms have the same performance with respect to the corresponding dimension (Note: optimizing global time for both discrete and continuous type of selection algorithms characterizing a *break-even point* reads on have the same performance with respect to the corresponding dimension or range – see Fig. 3.17, 3.18 pg. 80-81).

As per claim 8, Wu_D17 discloses computer program product for automatic software tuning comprising a plurality of instructions that when loaded into a memory of a computer system (Fig. 1.1, pg. 14) cause at least one processor of the computer system to execute the method of claim 1 (refer to claim 1).

As per claim 10, Wu_D17 discloses computer program product for dynamically selecting a data retriever implementation for retrieving data from a data storage system in response to a Boolean expression (sec 4.1, 4.2.2, 4.3 pg. 86-116 – Note: AND, OR operators using in relational operations and queries reads on boolean expression – see Appendix F, pg. 213), the computer program product comprising:

a result counter to determine a number of hits (result set of a selection – pg. 33; Fig. 2.6,2.7 pg. 37; result set – pg. 69-70, 91-92, 131-132, 135-136; Fig. 4.3, pg. 89; Fig. 4.12 pg. 107; page hits – pg. 125-126) in response to the Boolean expression;

a threshold evaluator to compare the number of hits (see *result set* - from above) with a threshold value of a first dimension ($A \leq 864$ – sec 4.2.1 – pg. 95; A op v – Algorithm 4 - pg. 97; significance threshold – pg. 115 – Note: comparing threshold or upper bound with *result set* or hits via a scenario applied to a selected range – see *RangeEval-Opt*, Appendix C, pg. 201-202; sec 3.5, pg 76; sec 4.2.2 pg. 96-97 - or bitmap indexing range scenario/algorithm reads on comparing hits with threshold associated with one dimension);

Wu_D17 discloses complexity of a boolean expression (Boolean expression – pg. 52-53, 56, 185; Appendix F) in terms of analyzing result from a range selection via indexing approach and responding to the algebraic constraint of the query having boolean operators (complexity – pg. 35, 62, 77, 96-97, 102), hence has disclosed complexity of the Boolean

expression, and this is indicative of applying analysis of Boolean expression or query tasks related time complexity in evaluating threshold or bitmap indexing alternatives with result set of running a range (see Appendix C pg. 201-202; sec 3.5, pg 76; sec 4.2.2 pg. 96-97).

Wu_D17 does not explicitly disclose compare the complexity of the Boolean expression with a further threshold value of a second dimension; but improving a range selection via modifying restrictions of a threshold via evaluating result and response time measurements related to scenarios driven by a range and threshold has been taught in Wu_D17 as set forth in claim 1 and this is equivalent to readjusting of upper bounds or range via compare the complexity of the Boolean expression with a further threshold value of a second dimension. Hence it would have been obvious for one skill in the art at the time the invention was made to implement the adjusting of threshold in Wu_D17 so that complexity analysis and related result set (i.e. hits related to a range selection) related to performance (response time collected from the scenario being run) comprise comparing the complexity of the Boolean expression (Boolean algebraic construct for a Aggregation, Selection or Group bys – see sec 4.1, pg. 86) with a further threshold value of a second dimension (evaluating a threshold in view of improving the upper limit via another range selection) because this would yield improved result set and the ensuing global performance as set forth in claim 1 given the overhead issues in applying too wide a range or upper bound (i.e. significance threshold θ) for collecting time/space optimization satisfactory results.

Nor does Wu_D17 explicitly disclose:

(i) a first data retriever to retrieve the data in case the *number of hits is below the threshold value of the first dimension and the complexity of the Boolean expression is above the further threshold value of the second dimension*;

(ii) a second data retriever to retrieve the data in case the *number of hits is above the threshold value of the first dimension and the complexity of the Boolean expression is above the further threshold value of the second dimension*;

(iii) a third data retriever to retrieve the data in case the *number of hits is below the threshold value of the first dimension and the complexity of the Boolean expression is below the further threshold value of the second dimension*; and

(iv) a forth data retriever to retrieve the data in case the *number of hits is above the threshold value of the first dimension and the complexity of the Boolean expression is below the further threshold value of the second dimension*.

The purpose of fulfilling a query is to retrieve data based on predicated requirement including the boolean expression requirement involving the upper and lower bounds range evaluation (Boolean expression - sec 4.1, 4.2.2, 4.3 pg. 86-116 – Note: AND, OR operators using in relational operations and queries reads on boolean expression and inequalities operators – see Appendix F, pg. 213); and since predicated expressions with inequalities operators include EQU, LT, GT, LEQ, GEQ – a concept inherent in boolean Algebra such as used in relational data warehouse query language by Wu_D17 - the comparing to match a range in terms of comparing to respectively a *equal to*, a *less than*, a *greater than*, a *less or equal to*, and *greater or equal to* constraints would be deemed necessary in the queries taught in Wu_D17's algorithms based on analyzing their complexity (Algorithm 2,3 pg. 77-78; Algorithm 4, 6-9, pg. 97-119).

It would have been obvious for one skill in the art at the time the invention was made to implement the analyzing of algorithms for tasks (Aggregation, selection, Group bys, Order bys) and related complexity -- as in the range evaluation dynamic approach by Wu_D17 -- so that data or hits retrieved in applying alternative query runs based on a given complexity and chosen threshold (sec 4.3.2, pg. 115-116) are compared with respect to the threshold and the complexity observed in using one dimension or another, where the comparing implement the inequalities (*equal to*, a *less than*, a *greater than*, a *less or equal to*, and *greater or equal)* evaluations in order to find out if the number of hits and threshold associated with a dimension are compared to a corresponding hits results and threshold associated with a second dimension to evaluate inequality matching as set forth in (i) (ii) (iii) (iv) because these comparing cases respectively implicate the obvious use of combination (inequalities LT+GT, GT + GT, LT+LT, GT+LT) evaluation deemed covering the result in and outside a range covered by a threshold, which Wu_D17 intends to improve as set forth in the range-evaluation incremental approach as set forth in claim 1, for the same benefits mentioned therein, which fall under the optimization of queries purported in Wu_D17's bitmap based static and dynamic selection approach (e.g. to retrieve data warehouse data).

As per claim 11, Wu_D17 discloses a retrieval time measuring component to measure the time that is consumed (response time – pg. 19-21, 107-108, 115-116; chp. 5 pg. 128-181; pg. 161-162, Appendix C pg. 201-202) by a selected data retriever for various numbers of hits; and (by virtue of the 103 rationale in claim 1) a threshold calculator to dynamically determine the threshold value and the further threshold value on the basis of the results of the retrieval time measuring component and to feed back the determined threshold values into the threshold

evaluator (refer to the rationale in claim 1 - to reevaluate threshold based on result feed backs into first run leading to modification of requirements, range changing and subsequent runs).

As per claims 12-13, Wu_D17 discloses wherein the first data retriever is implemented by using a general data retrieval algorithm using result flag instances (returnflag – Fig. 2.5 pg. 36; flags – pg 108-109); wherein the second data retriever is implemented by using a general data retrieval algorithm using bit maps (bitmaps – chp. 3-4, pg. 47-122).

As per claim s 14-15, Wu_D17 does not explicitly disclose wherein the third data retriever is implemented by using a *lean AND data* retrieval algorithm using result flag instances, wherein the forth[sic] data retriever is implemented by using a lean AND data retrieval algorithm using bit maps. But based on the alternative way Wu_D17 in readjusting the indexing approach (repartitioning a range and threshold – see Appendix C) based on response time and hits as set forth in claim 10, as well as the query complexity implicating predicate and boolean expression using restraints like a lean AND type of retrieval (And operator – pg. 33; AND logical operation – pg. 34), it would have been obvious for one skill in the art at the time the invention was made to implement the optimization to improve overhead and response time (e.g. from comparing first flag scenario with second scenario using bitmaps) in analyzing a *lean AND query* with adding or selecting a third scenario using flag instances so to compare its performance results with a scenario (a fourth scenario) using bit maps since the nature of complexity and the space size (see Fig. 2.10 pg 41) of data warehouse to consider would implicate which data retrieval (using a lean AND) indexing approach to utilize and according to Wu_D17, the bit map indexing approach as a emerged methodology (including encoded bitmap indexing) compared to conventional

algorithms would benefit of cooperativeness among bitmaps as well as iso/polymorphism associate with encapsulation of indexed structures (see: sec 2.3. 1 – pg. 33-34).

As per claim 16, refer to claim 10.

As per claim 17, Wu_D17 discloses a computer system for running a software application, the system comprising: variables for storing at least one threshold value (refer to claim 1) for at least one parameter (refer to claim 1) influencing the performance of the software application (refer to claim 1) with regards to a specific task (refer to claim 1); and a threshold evaluator for comparing(refer to claim 1) the at least one threshold value to at least one corresponding current value (refer to claim 1)

Wu_D17 does not disclose comparing explicitly in terms of *allowing the software application to select an algorithm from a plurality of algorithms for performing the task in accordance with the result of comparison*. However, this has been addressed as obvious using the rationale set forth in claim 1.

As per claims 18-23, refer to claims 2-7, respectively.

Conclusion

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (571) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lewis Bullock can be reached on (571)272-3759.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence - please consult Examiner before

using) or 571-273-8300 (for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Tuan A Vu/

Primary Examiner, Art Unit 2193

November 21, 2010